

**What Is Claimed Is:**

1. A computer system for generating output modules in a form-based application runtime environment, comprising:
  - a form manager component configured to receive an indication that a reusable form
  - 5 element has been changed, determine which output modules from a set of output modules are affected by the changed form element, and invalidate the affected output modules; and
  - a runtime manager component configured to receive a request for an output module from the set of output modules and cause regeneration of the requested output module if the requested output module has been invalidated.
- 10 2. The system of claim 1, wherein the indication is received when changes to the reusable form element are saved.
3. The system of claim 1, wherein the affected output modules are determined by referencing a
- 15 record data structure.
4. The system of claim 1, wherein the affected output modules are invalidated by marking a flag associated with each affected output module as invalid.
- 20 5. The system of claim 1, wherein the request for the output module received by the runtime manager is a request to identify the output module.
6. The system of claim 1, wherein the reusable form element is one of a form page and a form
- 25 window.
7. The system of claim 1, wherein the reusable form element is form logic.
8. The system of claim 1, wherein the reusable form element is a form interface.

~

9. A computer-implemented method for generating output modules in a form-based application runtime environment, comprising:
- receiving an indication that a reusable form element has been changed;
  - determining which output modules from a set of output modules are affected by the
  - 5 changed form element;
  - invalidating the affected output modules;
  - receiving a request for an output module from the set of output modules; and
  - regenerating the requested output module if the requested output module has been
  - invalidated.
- 10
10. The method of claim 9, wherein the indication is received when changes to the reusable form element are saved.
11. The method of claim 9, wherein the affected output modules are determined by referencing a
- 15 record data structure.
12. The method of claim 9, wherein the affected output modules are invalidated by marking a flag associated with each affected output module as invalid.
- 20 13. The method of claim 9, wherein the request for the output module received by the runtime manager is a request to identify the output module.
14. The method of claim 9, wherein the reusable form element is one of a form page and a form
- 25 window.
15. The method of claim 9, wherein the reusable form element is form logic.
16. The method of claim 9, wherein the reusable form element is a form interface.

3

17. A computer-implemented dynamic form building method, comprising:

responsive to a call to start a form output process based on an identified form:

determining whether a previously generated output module associated with the  
identified form in an output module library has been marked as invalid;

5 if so:

regenerating the output module; and

storing the regenerated output module in the output module library along with a  
marker to indicate that the output module is valid.

10 18. The method of claim 17, wherein the regeneration of the output module includes compiling  
changed reusable form elements into the output module.

24

19. A computer-implemented form library maintenance method, comprising:

upon revision to a form element, identifying from form element membership information

15 which forms from a form library are associated with the revised form element, and

marking each of the identified forms in the form library as invalid.